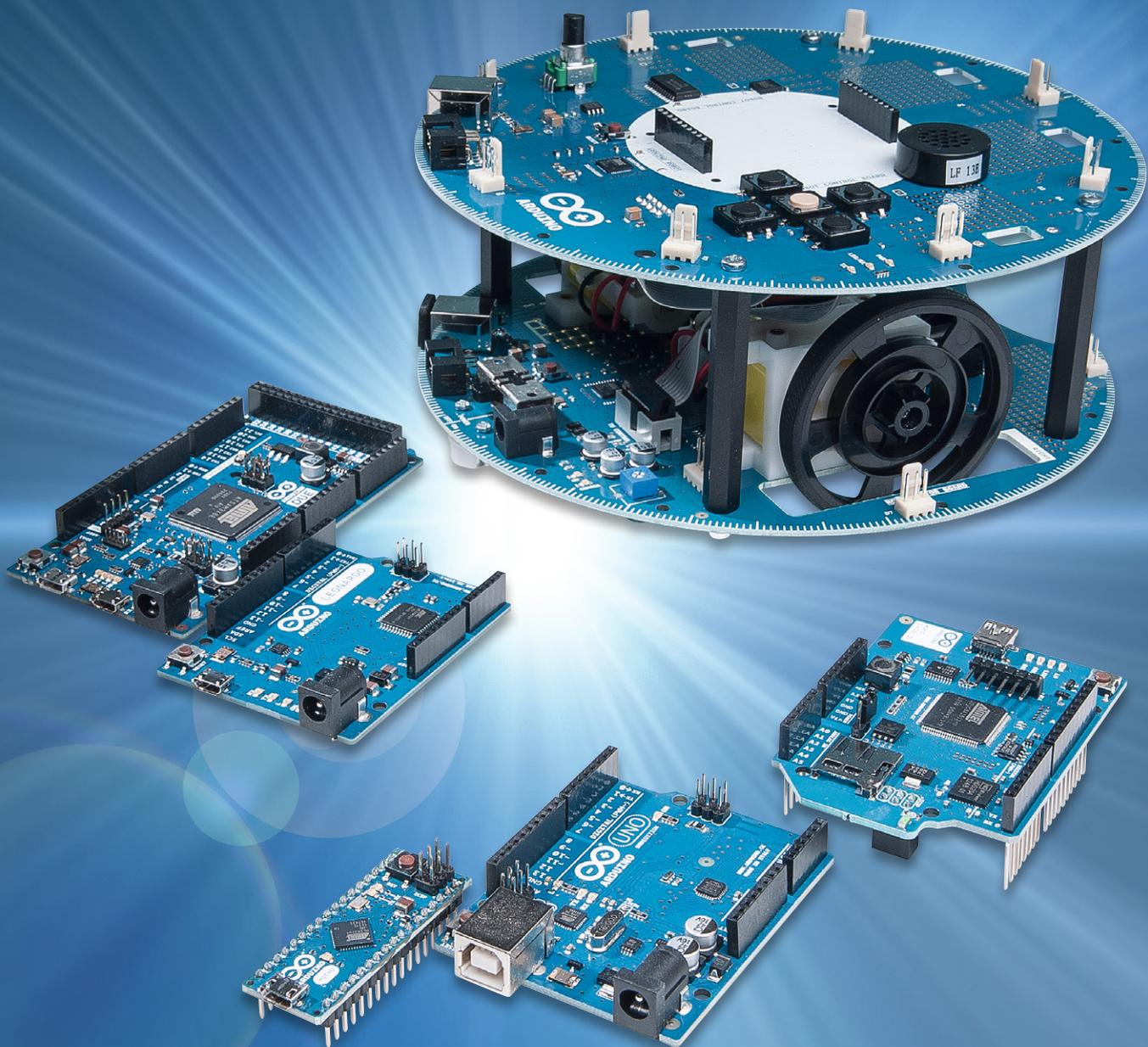




Arduino verstehen und anwenden

Teil 27: RFID-Technik für Haus und Heim





Berührungsloser Datenaustausch mit RFID-Technik

Die Abkürzung RFID steht für Radio Frequency Identification. Diese Technik ermöglicht es unter anderem, jeden Gegenstand, der mit einem RFID-Transponder ausgestattet ist, kontaktlos und eindeutig zu identifizieren. Das Wort „Transponder“ setzt sich aus den Bestandteilen „transmit“ und „response“ zusammen. Transponder können entweder nur einfache Ein-Bit-Informationen z. B. zur Diebstahlsicherung in Warenhäusern enthalten oder aber auch komplexere Daten übertragen. Im letzteren Fall kommuniziert ein integrierter Datenspeicher via Funk mit einem RFID-Lesegerät. Transponderkarten können entweder nur gelesen, in bestimmten Ausführungsarten aber auch beschrieben werden. Allerdings ist das zum Beschreiben nötige Verfahren relativ komplex, deshalb wird es hier nicht weiter verfolgt.

Die in diesem Artikel vorgestellten Anwendungen können auch mit den bereits in Standard-Transpondern vorhandenen Codes ausgeführt werden.

RFID-Lesegeräte sind in der Lage, von passiven Sendern, den sogenannten „RFID TAGs“, bestimmte Daten auszulesen. Damit lassen sich auch mit dem Arduino Schließanlagen oder ähnliche Projekte verwirklichen, bei denen sich eine Person mit einem TAG identifizieren soll.

Erste Entwicklungen zum Thema RFID begannen in den 60er-Jahren. Allerdings war die Technologie aufgrund hoher Herstellungskosten zunächst auf wenige Spezialanwendungen beschränkt. Diese Situation hat sich in den letzten Jahren grundlegend geändert. Inzwischen haben RFID-Verfahren in den verschiedensten Anwendungsgebieten ihren Platz gefunden. Rapide fallende Preise für RFID-Transponder und TAGs werden diesen Trend in Zukunft weiter verstärken.

Alle RFID-Systeme basieren auf einem einfachen Sender-Empfänger-Prinzip. Befindet sich der Transponder im Empfangsbereich des Lesegerätes, wird eine wechselseitige Kommunikation aufgebaut. Der Informationsaustausch erfolgt dabei über elektromagnetische Felder.

In den meisten Anwendungsfällen enthält ein RFID-System einen Sender, der genügend Energie abgibt, um den Empfänger in der Karte oder im Schlüsselanhänger zu aktivieren. Dieser sendet dann auf einer fest vorgegebenen Frequenz einen genau definierten Code zurück. Die Codesequenz wird dann wiederum empfangen und weiterverarbeitet. Wegen der Energieaufnahme aus dem Senderfeld benötigt ein sogenannter passiver TAG keine eigene Stromquelle. Er kommt also vollkommen ohne Batterie oder Akku aus und ist so dauerhaft einsatzfähig.

Durch den Einsatz von Transpondern mit internem Datenspeicher ergeben sich weiter interessante Anwendungsbereiche. Der Speicher erlaubt es, spezielle Informationen per RFID zu übermitteln. Ein beliebiger Gegenstand wird damit weltweit eindeutig identifizierbar. Zusätzlich können auch noch weitere Daten drahtlos auslesbar gemacht werden. Die aktuellen Speichergrößen in RFID-TAGs reichen von wenigen Bytes bis zu über 100 kB.

Ein RFID-Lesegerät besteht also aus einem Hochfrequenzmodul (Sender und Empfänger), einem Controller und einem Kopplungselement in

Form einer Spule, die gleichzeitig als Antenne dient. Diese Module sind in verschiedenen Varianten erhältlich. Für den Anschluss an den Arduino werden häufig Module eingesetzt, die über eine SPI-Schnittstelle gesteuert werden.

Energieversorgung von RFID-TAGs, Frequenzen und Reichweiten

Prinzipiell unterscheidet man RFID-Systeme, bei denen die Transponder ohne eigene Stromversorgung auskommen, sogenannte passive Systeme, und Transponder mit eigener Stromquelle, also aktive Varianten.

Ein einfacher TAG enthält einen auf das Empfangsgerät abgeglichenen Resonanzkreis. Außerhalb der Reichweite des RFID-Empfängers verhält sich der Transponder vollkommen passiv. Sobald er aber in die Nähe eines Detektors kommt, wird diesem durch resonante Übertragung eine minimale Energiemenge entnommen. Diese kann als binäre Information registriert werden. Darauf beruht der Einsatz von RFID-TAGs bei der Diebstahlsicherung in Kaufhäusern.

In Transpondern mit Datenspeicher befindet sich ebenfalls ein Schwingkreis, der auf die Frequenz des Senders im Lesegerät abgeglichen ist. Nähert sich der Transponder dem Wechselfeld, wird über dessen Spule eine für den Betrieb des Mikrochips ausreichende Energie in den Schwingkreis eingekoppelt. Steht genügend Energie zur Verfügung, sendet die Elektronik des Transponders den Inhalt des Datenspeichers über die Antenne an das Lesegerät zurück.

TAGs mit eigener Energieversorgung können auch noch arbeiten, wenn die übertragene Energie aufgrund der höheren Entfernung zur Versorgung des Chips nicht mehr ausreicht. Die empfangene Energie dient hier lediglich zur Erzeugung eines Wake-up-Signals. Die eigentliche Stromversorgung übernimmt die eingebaute Batterie.

Dem internen Aufbau entsprechend ergeben sich verschiedene Reichweiten von RFID-Systemen. Man unterscheidet drei Bereiche:

- **Close Coupling:** Hier werden Reichweiten von bis zu 1 cm erreicht. Die Transponder müssen in ein Lesegerät eingesteckt werden bzw. ihre Position muss beim Lesen genau definiert sein. Anwendungen, bei denen es auf höchste Sicherheit ankommt, werden durch dieses Verfahren realisiert.

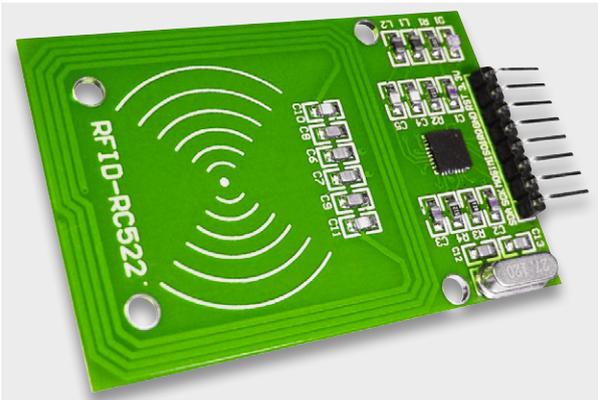


Bild 1: RFID-Modul

- **Remote Coupling:** Die Reichweiten bei diesen Systemen betragen bis zu 1 m, der Betrieb erfolgt bei Sendefrequenzen zwischen 100 und 135 kHz, 6,75 MHz, 13,56 MHz und 27,125 MHz. Die Energieversorgung des Transponders erfolgt passiv, d. h., zum Betrieb des Chips reicht die übertragene Energie aus dem Magnetfeld des Lesegerätes aus.
- **Long Range:** In diesem Bereich sind Reichweiten von 1 m bis zu 10 m typisch. Die Frequenzen liegen im Mikrowellenbereich. Die Energieübertragung reicht lediglich für die Sendung eines Wake-up- oder Sleep-Signals. Zur Versorgung des Transponderchips mit Energie und zur Erhaltung der gespeicherten Daten wird eine Stützbatterie benötigt.

Hinsichtlich der Frequenz können RFID-Systeme folgendermaßen eingeteilt werden:

- **Low Frequency** (30–500 kHz): Dieser Frequenzbereich ist überwiegend frei zugänglich und zeichnet sich durch geringe Übertragungsraten und Übertragungsabstände aus. Die Transponder arbeiten im Nahfeld des Senders und werden von diesem mit Energie versorgt.
- **High Frequency** (10–15 MHz): Dieses Band kann universell eingesetzt werden und zeichnet sich durch hohe Übertragungsraten aus. Besonders häufig kommt hier die Standardfrequenz von 13,56 MHz zum Einsatz.
- **Ultra High Frequency und Microwave-Bereich** (433 MHz, 850–950 MHz, 2,4–2,5 GHz): Hier werden die größten Reichweiten erzielt. Bei aktiven Transpondern mit eigener Energieversorgung liegen die Reichweiten bei über 30 Metern.



Bild 2: RFID-TAGs als Schlüsselanhänger und Keycard

Drahtlos Daten lesen: das RFID-Modul

Auch im nicht professionellen Bereich hat der Einsatz von RFID-Systemen in den letzten Jahren weite Verbreitung gefunden. Bild 1 zeigt ein Empfängermodul, das sich besonders gut zum Anschluss an einen Arduino eignet.

Dazu passend sind unterschiedliche TAGs erhältlich. Bild 2 zeigt zwei verschiedene Ausführungen: in Form eines Schlüsselanhängers und in Form einer Plastikkarte im Kreditkartenformat.

Neben diesen TAG-Formen existiert auch noch eine ganze Reihe weiterer Versionen. So findet man in Büchern oder Zeitschriften immer häufiger RFID-TAGs in Form von eingeklebten Etiketten. Auch diese TAGs können mit dem in Bild 1 gezeigten RFID-Empfänger ausgelesen werden.

Der Arduino liest RFID-TAGs

Die Kommunikation des RFID-Moduls mit dem Arduino erfolgt über das „Serial Peripheral Interface“ oder kurz SPI. Details zur Funktionsweise dieser Schnittstelle wurden im Artikel „SPI – Der serielle periphere Interface-Bus“ (ELV Journal 5/2017) eingehend erläutert. Diese Schnittstelle verfügt über drei Leitungen. Die Steuerleitung SCK (Serial Clock) wird vom Master zur Synchronisation mit einem frequenzstabilen Takt versorgt. Über MOSI (Master Output, Slave Input) und MISO (Master Input, Slave Output) werden die Daten gesendet bzw. empfangen. Dazu kommt noch für jeden Slave-Busteilnehmer eine Chip(Slave)-Select-Leitung (SS) und eventuell eine Reset-Leitung (RST). Die Interrupt-Leitung (IRQ) wird hier nicht verwendet. Die für die Verbindung von RFID-Empfänger und Arduino erforderlichen Signale sind in der folgenden Tabelle zusammengefasst:

RC522-Pin	Arduino-Pin
VCC (3V3)	3,3 V (!)
GND	GND
RST	9
SDA(SS)	10
MOSI	11
MISO	12
SCK	13

Dabei ist zu beachten, dass das Modul nur mit 3,3 V betrieben werden darf. Ein Anschluss an die 5-V-Versorgung des Arduinos würde das Modul überlasten.

Für das Auslesen des RFID-Empfängers kommt wieder eine vorgefertigte Bibliothek zum Einsatz. Diese kann unter <https://github.com/miguelbalboa/rfid> kostenlos aus dem Internet geladen werden.

In einem ersten Schritt kann man damit die sogenannte UID also die „Unique Identification Number“ auslesen. Diese enthält die individuellen Kenndaten eines RFID-TAGs. Das Programm dazu sieht so aus:

```
// RFID-RC522_UID.ino

#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN 9
#define SS_PIN 10

MFRC522 mfrc522(SS_PIN, RST_PIN);

void setup()
{ Serial.begin(9600);
  SPI.begin();
  mfrc522.PCD_Init();
```



```

Serial.println("Place RFID TAG in range!");
delay(300);
}

void loop()
{ if ( ! mfrc522.PICC_IsNewCardPresent())
  { return; }
  if ( ! mfrc522.PICC_ReadCardSerial())
  { return; }
  Serial.print("UID of RFID TAG: ");
  for (byte i = 0; i < mfrc522.uid.size; i++)
  { Serial.print(mfrc522.uid.uidByte[i], HEX);
    Serial.print(" ");
  }
  Serial.println();
}

```

```

COM5 (Arduino/Genuino Uno)
Place RFID TAG in range!
UID of RFID TAGS: 4E 8A 1E 2B

```

Bild 3: Ausgabe der UID im seriellen Monitor

Nach dem Starten des Programms kann eine Karte oder ein anderer TAG in die Reichweite des RFID-Moduls gebracht werden. Am seriellen Monitor sollte dann eine Ausgabe wie in [Bild 3](#) gezeigt erscheinen.

Im Programm werden nach dem Einbinden der erforderlichen Bibliotheken die Pins für die RST- und SS-Signale festgelegt. Die anderen Verbindungen entsprechen der Standardbelegung der SPI-Schnittstelle auf dem Arduino-Board und müssen daher nicht explizit angegeben werden. Anschließend wird der MFRC522-Prozess gestartet. Im Setup werden die RFID-Schnittstelle und das serielle Interface initialisiert. In der Hauptschleife erfolgt die Abfrage der Karte.

Falls kein RFID-TAG erkannt wird, bleibt das Ausgabefenster des seriellen Monitors leer. Sobald aber eine gültige Karte in die Nähe des Empfängers gehalten wird, liest das Modul die UID aus und leitet die Daten in hexadezimaler Form an die serielle Schnittstelle weiter.

RFID-Technik schafft Sicherheit

RFID-Systeme kommen häufig in der Zugangskontrolle zum Einsatz. In vielen Firmen und Instituten werden dann anstelle von konventionellen Schlössern RFID-Systeme verwendet. Die Mitarbeiter erhalten anstelle von Schlüsseln eine TAG-Karte und damit Zugang zu bestimmten Räumen. Dieses Vorgehen hat gegenüber den herkömmlichen Schlössern mit passenden Schlüsseln verschiedene Vorteile. So kann bei Verlust einer Karte einfach der zugehörige Code aus dem System genommen werden. Ein kostspieliges Ersetzen von Schließzylindern entfällt. Zudem kann Personen der Zugang verwehrt werden, auch wenn diese noch im Besitz eines TAGs sind. Ein einfaches Sperren der betreffenden ID genügt, um diese kurzzeitig oder auch dauerhaft am Betreten sicherheitsrelevanter Räume zu hindern. Ein weiterer Vorteil ist, dass TAGs wesentlich preisgünstiger sind als spezielle Sicherheitsschlüssel.

Auch das RFID-Modul am Arduino kann als Kontrollsystem verwendet werden. Für die Erstellung von gültigen Codes genügt es, die UIDs in Form einer sechsstelligen Dezimalzahl einzugeben. Das folgende Programm liefert zu jeder Karte einen entsprechenden individuellen Code:

```

// RFID-RC522_Code.ino

#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN 9
#define SS_PIN 10

MFRC522 mfrc522(SS_PIN, RST_PIN);

long code = 0;
long validCode = 123456; // enter valid code
int activePin = 7;

void setup()
{ pinMode (activePin, OUTPUT);
  Serial.begin(9600);
  SPI.begin();
  mfrc522.PCD_Init();
  Serial.println("Place RFID TAG in range!");
  delay(300);
}

void loop()
{ if ( ! mfrc522.PICC_IsNewCardPresent())
  { return; }
  if ( ! mfrc522.PICC_ReadCardSerial())
  { return; }

  Serial.print("UID of RFID TAGS: ");
  for (byte i = 0; i < mfrc522.uid.size; i++)
  { Serial.print(mfrc522.uid.uidByte[i], HEX);
    Serial.print(" ");
  }
  Serial.println();

  long code=0;
  for (byte i = 0; i < mfrc522.uid.size; i++)
  {code=((code+mfrc522.uid.uidByte[i])*10);
  }
  Serial.print("Code is: ");
  Serial.println(code);
  if (code == validCode)
  { Serial.println("Valid ID detected - activate lock!");
    Serial.println();
    digitalWrite (activePin, HIGH);
  }
}

```



```
delay (3000);  
digitalWrite (activePin, LOW);  
}  
}
```

Man muss nun nur noch die Codesequenz derjenigen TAGs auswählen, die als Schlüssel dienen sollen. Diese werden als zugangsberechtigte Karten-UIDs im Sketch eingetragen, z. B.:

```
long validCode = 123456;
```

Die Karte mit der ID 123456 wird dann als passender Schlüssel betrachtet. Wird sie in die Nähe des Empfängermoduls gehalten, wird der „activePin“ (Port 7) für drei Sekunden auf den Wert „HIGH“ geschaltet.

So kann man beispielsweise einen elektromagnetischen Türriegel steuern. Bei allen anderen Karten bleibt der Port dagegen im LOW-Zustand und die zugehörige Türverriegelung wird nicht freigegeben.

Chipkarten als Informationsträger

RFID-TAGs können auch vergleichsweise große Datenmengen aufnehmen. Diese können ebenfalls mit dem Arduino ausgelesen werden. Der nachfolgende Sketch ermöglicht es dem RFID-Modul, diese Daten zu empfangen und an die serielle Schnittstelle des Arduino UNOs weiterzuleiten.

TAGs mit größeren Dateninhalten werden häufig zu Werbezwecken eingesetzt. So kann man etwa Internetadressen auf den kostengünstigen Karten speichern. Diese werden dann z. B. auf Messen, Ausstellungen oder Verkaufsveranstaltungen verteilt. So erreicht man mit geringem finanziellen Aufwand einen großen potenziellen Kundenkreis. Aber auch in Büchern oder Zeitschriften können TAGs in Form von Aufklebern verbreitet werden. RFID-Empfänger können die Daten lesen, und das aufwendige und fehlerträchtige Eintippen von Internetadressen entfällt. Der folgende Sketch erlaubt es, solche RFID-Etiketten auszuwerten:

```
// RFID-RC522_data.ino  
  
#include <SPI.h>  
#include <MFRC522.h>  
  
#define RST_PIN      9  
#define SS_PIN       10  
  
MFRC522 mfrc522(SS_PIN, RST_PIN);  
  
void setup()  
{ Serial.begin(9600);  
  SPI.begin();  
  mfrc522.PCD_Init();  
  mfrc522.PCD_DumpVersionToSerial();  
  Serial.println("Place RFID TAG in range!");  
}  
  
void loop()  
{ if ( ! mfrc522.PICC_IsNewCardPresent())  
  { return; }  
  if ( ! mfrc522.PICC_ReadCardSerial())  
  { return; }  
  mfrc522.PICC_DumpToSerial(&(mfrc522.uid));  
}
```

Sobald ein TAG-Aufkleber in den Empfangsbereich des Moduls gehalten wird, erscheinen die darauf gespeicherten Daten auf dem seriellen Monitor der Arduino-IDE.

RFID-gesicherte Tür oder Lockbox

Abschließend soll in einem Anwendungsbeispiel die Steuerung eines elektronischen Schlosses über einen RFID-Chip vorgestellt werden. Softwareseitig sind hierfür drei Bibliotheken erforderlich. Zum einen die SPI-Lib aus der Arduino-IDE, dann die MFRC522-Lib für das RFID-Modul und schließlich die Library zur Ansteuerung eines Modellbauservos.

Wenn die Hardware nach Bild 4 aufgebaut ist, kann der Sketch „RFID-RC522_data“ aus dem Download-Paket auf den Arduino geladen werden. Damit lassen sich bereits Daten aus einem RFID-Transponder lesen, wenn man den seriellen Monitor der Arduino-IDE öffnet. Sobald ein Transponder in die Nähe des RFID-Moduls gehalten wird, erscheinen nun die auf dem Chip gespeicherten Daten.

Nach der Eingabe eines gültigen Codes in der Zeile `long validCode = 123456; // enter valid code` ist das System einsatzbereit.

Zum Ansteuern der Schließmechanik kommt ein Modellbauservo zum Einsatz. Will man den Inhalt eines Behälters vor unbefugtem Zugriff schützen, so kann man das System als „Lockbox“-Schloss einsetzen. Bild 5 zeigt einen entsprechenden Aufbauvorschlag. Das vollständige System aus Arduino, RFID-Modul und Servo kann problemlos in einer passenden Holzbox untergebracht werden. Für die Energieversorgung des Arduinos kommen noch geeignete Batterien dazu.

In der Box wird ein Verschlusshaken angebracht. Dieser muss so gebogen werden, dass der Servoarm in den Haken greift und die Box so sicher verschließt. Das RFID-Modul wird in der Nähe des Deckels angebracht. Nachdem die Box geschlossen wurde, kann sie nur mittels eines passenden RFID-TAGs wieder geöffnet werden.

Die grüne und eine rote LED können bei Bedarf über passende Löcher in den Deckel eingelassen werden. Für die Funktion der Box sind sie allerdings nicht unbedingt erforderlich. Wird die Box über längere Zeiträume hinweg verwendet, empfiehlt es sich, eine externe Stromversorgungsbuchse vorzusehen, da sich der Deckel bei entladenen Akkus bzw. Batterien sonst nicht mehr öffnen lässt.

Will man Lockboxen einsetzen, die bei Versagen des RFID-Systems nur mit hohem Aufwand aufgebrochen werden können, empfiehlt es sich, einen Notfallmechanismus einzubauen, der das Öffnen der Box auch bei defektem RFID-System, z. B. mit einem Notfallschlüssel, ermöglicht. Ansonsten besteht das Risiko, dass die Box, etwa bei Versagen eines elektrischen Kontaktes, nicht mehr geöffnet werden kann.



Ausblick

In diesem Beitrag wurden die Grundlagen der RFID-Technik erläutert und entsprechende Anwendungen im Arduino-Umfeld vorgestellt. Damit ist der Exkurs in den Bereich von drahtlosen Datenübertragungssystemen in Verbindung mit dem Arduino abgeschlossen.

Im nächsten Artikel geht es um die verschiedenen Arduino-Varianten, die inzwischen auf dem Markt erhältlich sind. Neben den kleinen Brüdern des Arduino UNO, dem MIKRO und NANO werden auch die großen Varianten wie DUE und MEGA vorgestellt. Zum Abschluss wird gezeigt, wie Arduinos im Eigenbau hergestellt werden können. Damit lässt sich dann „Arduino-Hardware“ auch an ganz spezielle Erfordernisse anpassen. **ELV**



Weitere Infos:

- Grundlagen zur elektronischen Schaltungstechnik finden sich in der E-Book-Reihe „Elektronik!“ (www.amazon.de/dp/B000XNCB02)
- Elektor-Praxiskurs AVR-XMEGA-Mikrocontroller, Bestell-Nr. CU-12 07 62
- Franzis Physical Computing, Bestell-Nr. CU-12 21 81
- Franzis Lernpaket Motoren & Sensoren mit Arduino, Bestell-Nr. CU-12 74 74

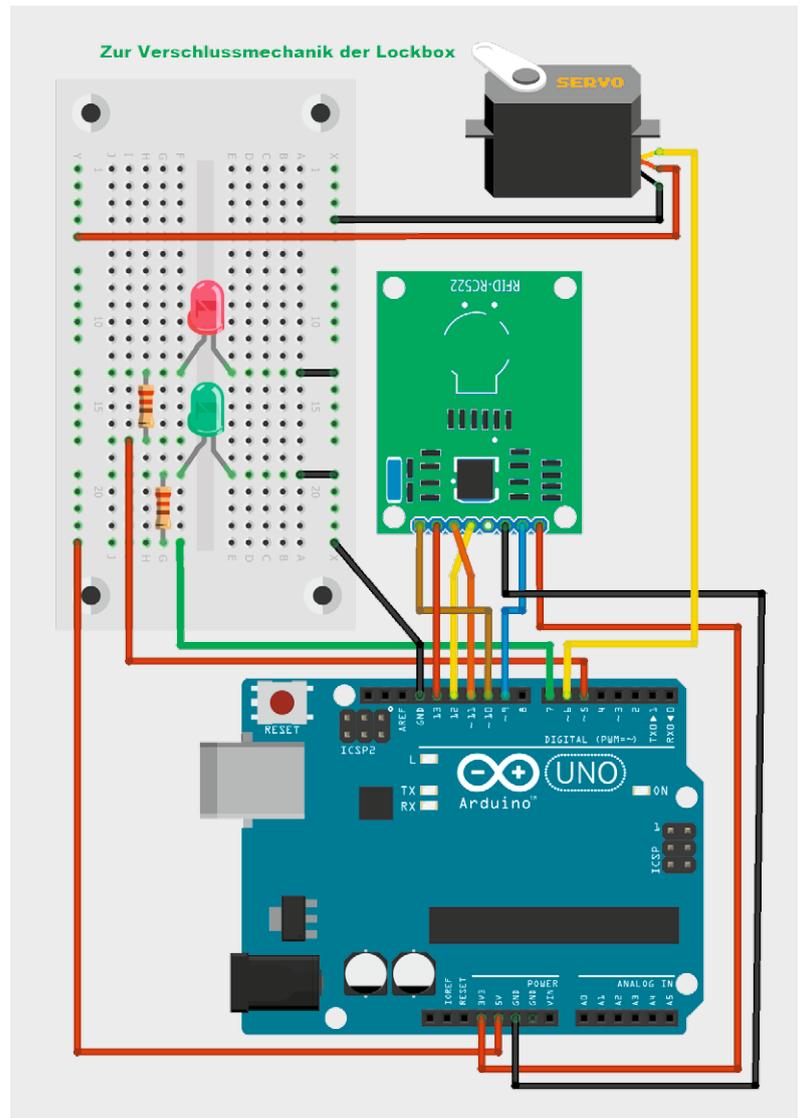


Bild 4: Servosteuerung via RFID

Download-Paket zum Artikel:

Die Sketche und Beispieldateien zu diesem Artikel können kostenlos heruntergeladen werden unter www.elv.de: Webcode #10180

Preisstellung Februar 2018 – aktuelle Preise im ELV Shop

Empfohlene Produkte	Best.-Nr.	Preis
Arduino UNO	CU-10 29 70	€ 27,95
Joy-iT RFID-Modul basierend auf NXP MFRC-522	CU-12 75 17	€ 9,95
RFID-Transponder Schlüsselanhänger, 10er-Pack	CU-10 79 27	€ 9,95

Alle Arduino-Produkte wie Mikrocontroller-Platinen, Shields, Fachbücher und Zubehör finden Sie unter: www.arduino.elv.de

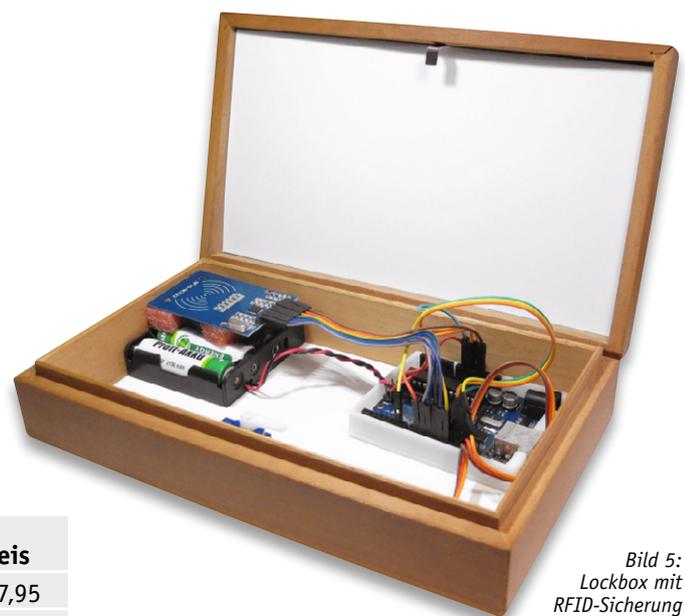


Bild 5: Lockbox mit RFID-Sicherung